```
836 Increment GoalMet count
838 If (Saved probation is true)  Then
 840 GoalMetMultiplier =1 (forget previous bad history, good sample after probation)
End
842 If (GoalMet > GoalMetMultiplier *4)  Then
 844 GoalMet = 0
 846 If (DelayInterval =0)  Then
  848 Set DelayInterval to target delay interval constant
 End
End
850 End
```

[0050] A level of collaboration between the computer hypervisor implementing the floating CPU partitioning, and the devices is required. This collaboration involves the hypervisor proactively informing the devices when the target partition is no longer active on any CPU (i.e. dispatcher polling is not occurring for that partition). This proactive notification would cause the devices to generate an interrupt immediately, independent of the current delay value calculation.

[0051] Logical partitioning implementations require the hypervisor to track which CPUs are all allocated to a given logical partition for a number of reasons. One such reason is to be able to detect when the last CPU is removed from a logical partition, so that the hypervisor can inform the hardware that interrupts targeted for that partition must now be handled by the hypervisor instead of the logical partition itself. In the preferred implementation a hook is added to this interrupt redirection processing, to proactively inform devices that an immediate interrupt will be required to cause the activation of the target partition. Then as part of processing that interrupt, or the logical partition performing a dispatcher polling activity (which ever comes first), the hypervisor notice is reset.

[0052] Turning now to Fig. 15, the computer 210 is shown divided, for example, into four partitions, 710, 712, 714, and 716, as is well known. A hypervisor 720 oversees the partitions,

and assigns which of the CPUs 130 run in each of the partitions, as is well known. In the example shown in Fig. 13, partitions 1, 2 and 4 (710, 712 and 716) have CPUs assigned which perform the polling previously described and as represented by 724, 726, and 728. As an example, partition 3 (714) has had its CPU removed. In this case, the hypervisor 720 informs the hardware that I/O interrupts will be handled by the hypervisor 720 rather than using the polling techniques described. Thus, when a device 190 requests an I/O operation with partition 3 (714) the hypervisor 720 handles an immediate interrupt, as will be described.

[0053] Referring to Fig. 16, each partition of the computer 210 has an override bit 730 associated with the GSB 612 for that partition. When the partition does not have a CPU assigned, or when a CPU is removed from the partition, the hypervisor 720 sets the override bit 730 to inform any devices 190 requesting an I/O operation with the partition as shown at 732, that an immediate interrupt should be handled by the hypervisor 720. When the hypervisor 720 processes the interrupt, or when a CPU is reassigned to the partition and a dispatcher performs a polling activity for that partition, whichever occurs first, the override bit is reset as shown at 734.

[0054] It will be understood that even though the example of Fig. 14 is with a machine divided into four partitions, the override bit will be the same for a machine divided into any number of partitions, or in a machine not divided into partitions where it is desirable to notify the hardware that an immediate interrupt should be taken rather than polling the hierarchy.

[0055] While the preferred embodiment of the invention has been illustrated and described herein, it is to be understood that the invention is not limited to the precise construction herein disclosed, and the right is reserved to all changes and modifications coming within the scope of the invention as defined in the appended claims.